Domain Adaptation using Passive and Active Approaches

Dr Haider Raza

Postdoctoral Research Fellow
Institute for Analytics and Data Science, University of Essex,
Colchester, Essex, UK.
h.raza@essex.ac.uk

: @sagihaider
Linkedin : sagihaider

"**10th Computer Science and Electronic Engineering Conference**"

19^{th} Sept 2018

GitHub: Lab work and presentation

Follows the steps:

- Go to link https://github.com/sagihaider/CEEC_2018
- Look at right hand side in green color: Clone or download. Click it and Download Zip
- When downloading is finished. Copy the Zip file and take to the location you want such any folder and paste it. Extract it.
- If you have **Anaconda3** installed. Go to terminal or command prompt and type "jupyter notebook"

Outline

Shifts in Data

- Dataset Shift
- Types of Dataset Shift
- Causes of Dataset Shift

Learning in Dataset Shift: **Domain Adaptation**

Notation

- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or $P(Y \cap X)$ (i.e. Probability of Y and X).
- $(X \rightarrow Y)$: Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- $(Y \rightarrow X)$: Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
  1. $P(Y|X)P(X)$ in $X \rightarrow Y$ problems.
  2. $P(X|Y)P(Y)$ in $Y \rightarrow X$ problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Notation

- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or P(Y $\cap$ X) (i.e. Probability of Y and X).
- (X $\rightarrow$ Y): Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- (Y $\rightarrow$ X): Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
    1. P(Y|X)P(X) in X $\rightarrow$ Y problems.
    2. P(X|Y)P(Y) in Y $\rightarrow$ X problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Notation

- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or P(Y ∩ X) (i.e. Probability of Y and X).
- (X → Y): Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- (Y → X): Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
  1. P(Y|X)P(X) in X → Y problems.
  2. P(X|Y)P(Y) in Y → X problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Notation

- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or P(Y $\cap$ X) (i.e. Probability of Y and X).
- (X $\rightarrow$ Y): Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- (Y $\rightarrow$ X): Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
  1. P(Y|X)P(X) in X $\rightarrow$ Y problems.
  2. P(X|Y)P(Y) in Y $\rightarrow$ X problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Notation

- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or P(Y $\cap$ X) (i.e. Probability of Y and X).
- (X $\rightarrow$ Y): Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- (Y $\rightarrow$ X): Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
  1. P(Y|X)P(X) in X $\rightarrow$ Y problems.
  2. P(X|Y)P(Y) in Y $\rightarrow$ X problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Notation

- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or P(Y $\cap$ X) (i.e. Probability of Y and X).
- (X $\rightarrow$ Y): Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- (Y $\rightarrow$ X): Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
  1. P(Y|X)P(X) in X $\rightarrow$ Y problems.
  2. P(X|Y)P(Y) in Y $\rightarrow$ X problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Notation

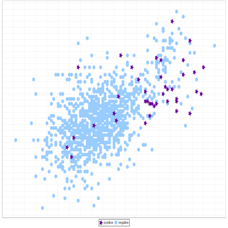
- A set of features or covariates X.
- A set of target or class variables Y.
- A joint distribution P(Y,X) or $P(Y \cap X)$ (i.e. Probability of Y and X).
- $(X \rightarrow Y)$: Y is determined by values of X (e.g. credit card fraud detection) Predictive models (e.g. Logistic Regression, SVM, and Neural Networks.)
- $(Y \rightarrow X)$: Y determines the values of X (e.g. medical diagnosis) Generative models (e.g. GMM, HMM, and Naive Bayes).
- The joint distribution P(Y,X) can be written as
  1. $P(Y|X)P(X)$ in $X \rightarrow Y$ problems.
  2. $P(X|Y)P(Y)$ in $Y \rightarrow X$ problems.
- P_{tr}: Data distribution in training
- P_{ts}: Data distribution in testing

Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

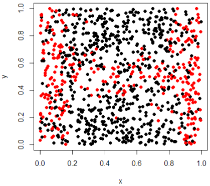


Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

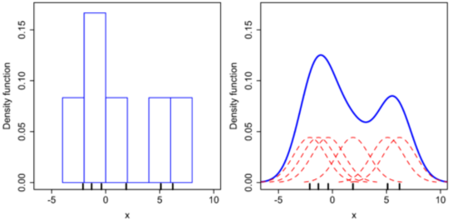


Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

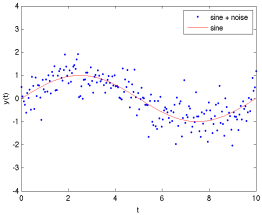


Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

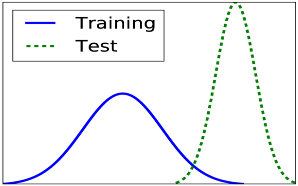


Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift

Issues with Data in Machine Learning

- Imbalanced dataset
- Overlapping dataset
- Density: Lack of data
- Noise in data
- Dataset Shift



Motivation

- In learning theory independent and identically distributed (i.i.d) assumption (i.e. each random variable has the same probability distribution as the others and all are mutually independent).
- In practice train and test inputs have different distributions.
- The difference in distribution arises from operating in non-stationary environments in real-world application such as finance, healthcare, brain signals, much more...
- Learning in such non-stationary environment is difficult and we need an think before operating.

Motivation

- In learning theory independent and identically distributed (i.i.d) assumption (i.e. each random variable has the same probability distribution as the others and all are mutually independent).
- In practice train and test inputs have different distributions.
- The difference in distribution arises from operating in non-stationary environments in real-world application such as finance, healthcare, brain signals, much more...
- Learning in such non-stationary environment is difficult and we need an think before operating.

Motivation

- In learning theory independent and identically distributed (i.i.d) assumption (i.e. each random variable has the same probability distribution as the others and all are mutually independent).
- In practice train and test inputs have different distributions.
- The difference in distribution arises from operating in non-stationary environments in real-world application such as finance, healthcare, brain signals, much more...
- Learning in such non-stationary environment is difficult and we need an think before operating.

Motivation

- In learning theory independent and identically distributed (i.i.d) assumption (i.e. each random variable has the same probability distribution as the others and all are mutually independent).
- In practice train and test inputs have different distributions.
- The difference in distribution arises from operating in non-stationary environments in real-world application such as finance, healthcare, brain signals, much more...
- Learning in such non-stationary environment is difficult and we need an think before operating.

Shifts in Data

Outline

Dataset Shift

- "cases where the joint distribution of inputs and outputs differs between training and test stage"[1]

1. "concept shift/drift" G. Widmer et al., 1996, 1998
2. "changes of classification" K. Wang et al., 2003
3. "changing environments" R. Alaiz-Rodrıguez et al., 2008
4. "fracture point" N.V. Chawla et al., 2009
5. "fractures between data" J.G. Moreno-Torres et al., 2010

[1] *A. Storkey, Dataset Shift in Machine Learning, 2009*

Dataset Shift

- "cases where the joint distribution of inputs and outputs differs between training and test stage"[1]
  1. "concept shift/drift" G. Widmer et al., 1996, 1998
  2. "changes of classification" K. Wang et al., 2003
  3. "changing environments" R. Alaiz-Rodrıguez et al., 2008
  4. "fracture point" N.V. Chawla et al., 2009
  5. "fractures between data" J.G. Moreno-Torres et al., 2010

[1] A. Storkey, Dataset Shift in Machine Learning, 2009

Dataset Shift

- "cases where the joint distribution of inputs and outputs differs between training and test stage"[1]
  1. "concept shift/drift" G. Widmer et al., 1996, 1998
  2. "changes of classification" K. Wang et al., 2003
  3. "changing environments" R. Alaiz-Rodrıguez et al., 2008
  4. "fracture point" N.V. Chawla et al., 2009
  5. "fractures between data" J.G. Moreno-Torres et al., 2010

[1] A. Storkey, Dataset Shift in Machine Learning, 2009

Dataset Shift

- "cases where the joint distribution of inputs and outputs differs between training and test stage"[1]
  1. "concept shift/drift" G. Widmer et al., 1996, 1998
  2. "changes of classification" K. Wang et al., 2003
  3. "changing environments" R. Alaiz-Rodrıguez et al., 2008
  4. "fracture point" N.V. Chawla et al., 2009
  5. "fractures between data" J.G. Moreno-Torres et al., 2010

[1] A. Storkey, Dataset Shift in Machine Learning, 2009

Dataset Shift

- "cases where the joint distribution of inputs and outputs differs between training and test stage"[1]
  1. "concept shift/drift" G. Widmer et al., 1996, 1998
  2. "changes of classification" K. Wang et al., 2003
  3. "changing environments" R. Alaiz-Rodrıguez et al., 2008
  4. "fracture point" N.V. Chawla et al., 2009
  5. "fractures between data" J.G. Moreno-Torres et al., 2010

[1] *A. Storkey, Dataset Shift in Machine Learning, 2009*

Dataset Shift

- "cases where the joint distribution of inputs and outputs differs between training and test stage"[1]
  1. "concept shift/drift" G. Widmer et al., 1996, 1998
  2. "changes of classification" K. Wang et al., 2003
  3. "changing environments" R. Alaiz-Rodrıguez et al., 2008
  4. "fracture point" N.V. Chawla et al., 2009
  5. "fractures between data" J.G. Moreno-Torres et al., 2010

[1] *A. Storkey, Dataset Shift in Machine Learning, 2009*

Dataset Shift: General Example

- Speech recognition system
- Training the speech recognition system
- Voice recognition systems fails

Dataset Shift: General Example

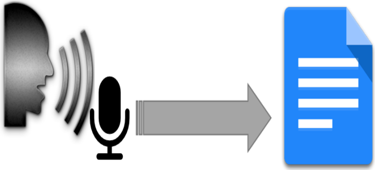
- Speech recognition system
- Training the speech recognition system
- Voice recognition systems fails

Dataset Shift: General Example

- Speech recognition system
- Training the speech recognition system
- Voice recognition systems fails

Dataset Shift: General Example

- Speech recognition system
- Training the speech recognition system
- Voice recognition systems fails

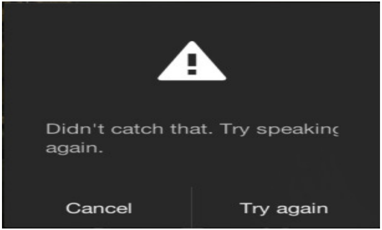


Dataset Shift: General Example

- Speech recognition system
- Training the speech recognition system
- Voice recognition systems fails

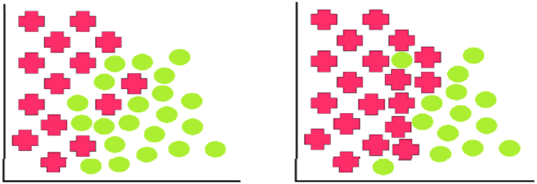
Dataset Shift: General Example

- Speech recognition system
- Training the speech recognition system
- Voice recognition systems fails



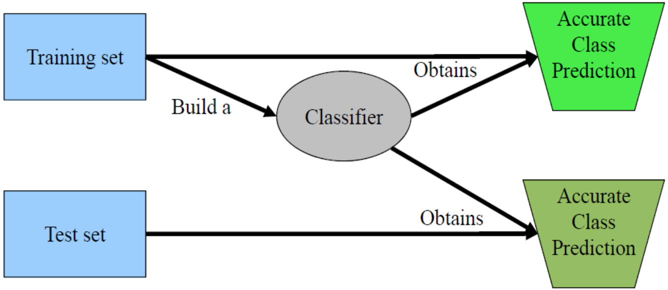
Dataset Shift...cont

- **Dataset shift** appears when training and test joint distributions are different. That is, when $P_{tr}(X,Y) \neq P_{ts}(X,Y)$



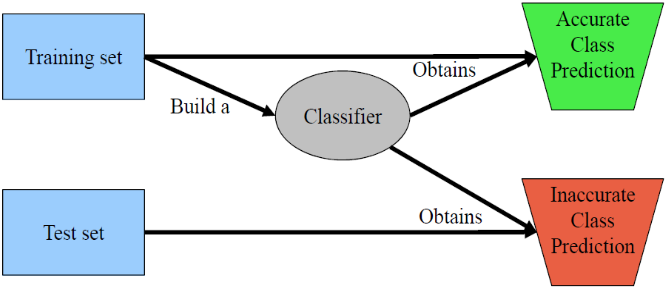
Dataset Shift...cont

- Basic assumption for classification in operating under stationary environment



Dataset Shift...cont

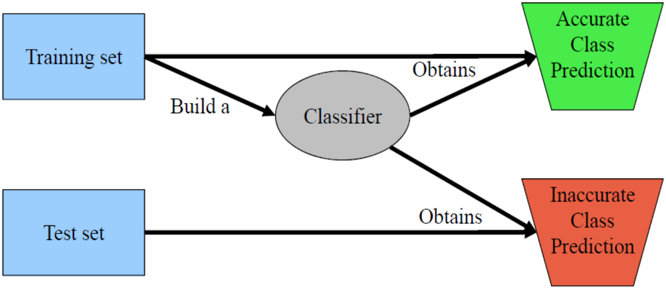
- But sometimes...



- The classifier has overfitting problem

Dataset Shift...cont

- But sometimes...



- The classifier has overfitting problem

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] *Alippi et al., IEEE TNNLS, 2008.*

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] *Alippi et al., IEEE TNNLS, 2008.*

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] *Alippi et al., IEEE TNNLS, 2008.*

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] *Alippi et al., IEEE TNNLS, 2008.*

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] *Alippi et al., IEEE TNNLS, 2008.*

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] Alippi et al., IEEE TNNLS, 2008.

The problem of Dataset Shift

- **If** the classifier has an overfitting problem: **then** possible actions
  - Change the parameters of the algorithm
  - Use a more general learning method
- **If** there is a change in the data distribution between training and test sets: **then** possible actions [2]
  - Train a new classifier for the test set
  - Adapt to classifier
  - Modify the data in the test set

[2] *Alippi et al., IEEE TNNLS, 2008.*

Outline

Types of dataset shift

1. Covariate shift
2. Prior probability shift
3. Concept shift Concept

Types of dataset shift

1. Covariate shift
2. Prior probability shift
3. Concept shift Concept

Types of dataset shift

1. Covariate shift
2. Prior probability shift
3. Concept shift Concept

Covariate Shift

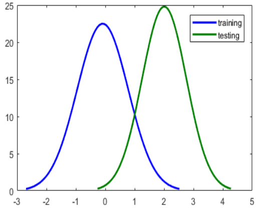
- Covariate shift appears only in $X \rightarrow Y$ problems [3], and is defined as the case where

$$P_{tr}(Y \mid X) = P_{ts}(Y \mid X)$$

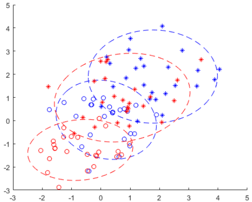
$$\&$$

$$P_{tr}(X) \neq P_{ts}(X)$$

Uni-variate



Bi-variate



[3] *Raza et al., Pattern Recognition, 2015.*

Covariate Shift

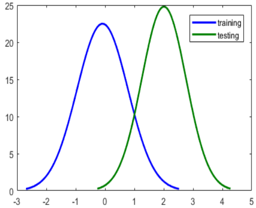
- Covariate shift appears only in $X \rightarrow Y$ problems [3], and is defined as the case where

$$P_{tr}(Y \mid X) = P_{ts}(Y \mid X)$$

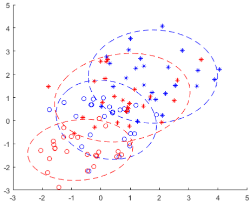
$$\&$$

$$P_{tr}(X) \neq P_{ts}(X)$$

Uni-variate



Bi-variate



[3] *Raza et al., Pattern Recognition, 2015.*

Covariate Shift

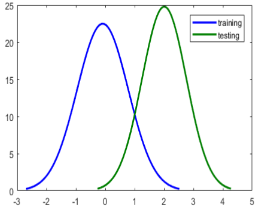
- Covariate shift appears only in $X \rightarrow Y$ problems [3], and is defined as the case where

$$P_{tr}(Y \mid X) = P_{ts}(Y \mid X)$$

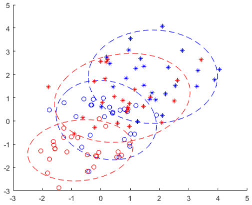
$$\&$$

$$P_{tr}(X) \neq P_{ts}(X)$$

Uni-variate



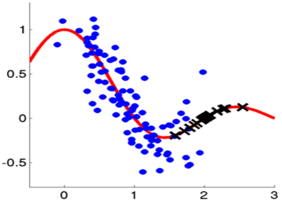
Bi-variate



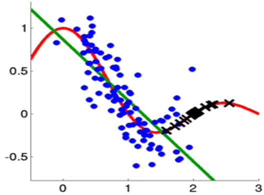
[3] *Raza et al., Pattern Recognition, 2015.*

Covariate Shift: An Example

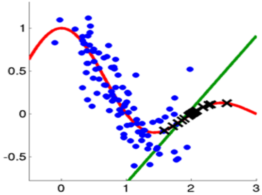
- A regression example [4]



Training



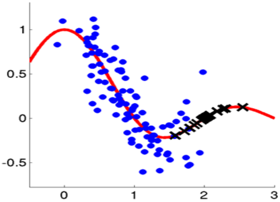
Testing



[4] *Sugiyama et al., Journal of Machine Learning Research, 2007.*

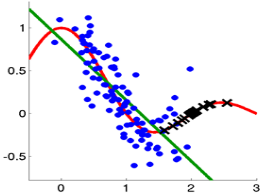
Covariate Shift: An Example

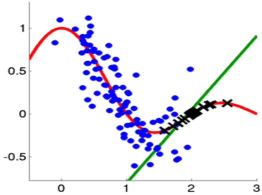
- A regression example [4]



Training

Testing

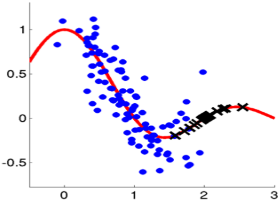




[4] *Sugiyama et al., Journal of Machine Learning Research, 2007.*

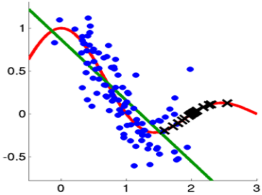
Covariate Shift: An Example

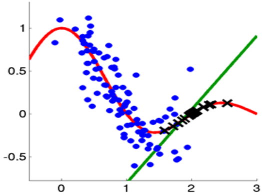
- A regression example [4]



Training

Testing

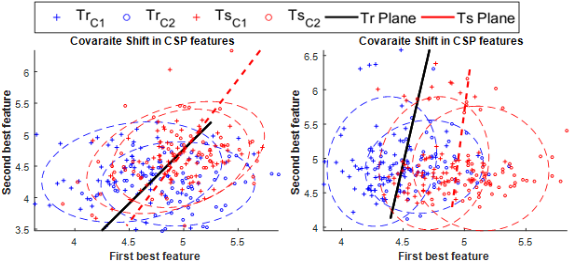




[4] *Sugiyama et al., Journal of Machine Learning Research, 2007.*

Covariate Shift: An Example

- A Classification example [5]

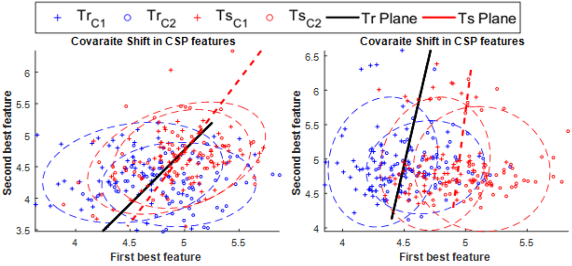


- Covariate shift (CS) between the training and test distributions of the EEG signal from the healthy subject (a) illustrates the CS in the mu band [8-12] Hz and (b) shows the CS in the beta band [14-30] Hz.

[5] Raza et al., Soft Computing .,2015 and IEEE IJCNN, 2015.

Covariate Shift: An Example

- A Classification example [5]



- Covariate shift (CS) between the training and test distributions of the EEG signal from the healthy subject (a) illustrates the CS in the mu band [8-12] Hz and (b) shows the CS in the beta band [14-30] Hz.

[5] *Raza et al., Soft Computing .,2015 and IEEE-IJCNN., 2015.*

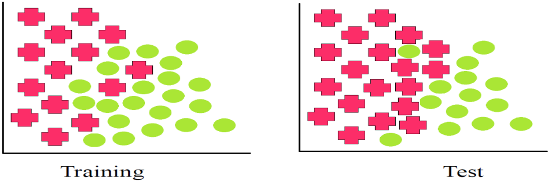
Prior probability shift

- Prior probability shift appears only in Y $\rightarrow$ X problems, and is defined as the case where

$$P_{tr}(Y \mid X) = P_{ts}(Y \mid X)$$

$$\&$$

$$P_{tr}(Y) \neq P_{ts}(Y)$$



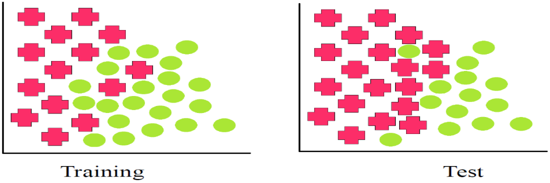
Prior probability shift

- Prior probability shift appears only in Y $\rightarrow$ X problems, and is defined as the case where

$$P_{tr}(Y \mid X) = P_{ts}(Y \mid X)$$

$$\&$$

$$P_{tr}(Y) \neq P_{ts}(Y)$$



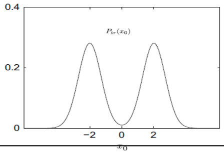
Prior probability shift: Example

Example [6]: $Y \rightarrow X$ problem with one covariate x_0 and a target y that may take the class value $y{=}0$ and $y{=}1$. In training data, $P_{tr}(y=0) = P_{tr}(y=1) = 0.5$ and $P_{tr}(x_0 \mid y)$ is defined as

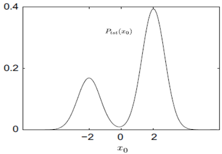
$$x_0 = \begin{cases} \mathcal{N}(2, 0.5), & \text{when } y = 1 \\ \mathcal{N}(-2, 05), & \text{otherwise} \end{cases} \quad (1)$$

Now consider that in the test data $P_{ts}(x_0 \mid y = 0)$ and $P_{ts}(x_0 \mid y = 1)$ remains unchanged, but the class prior probabilities vary, taking the values $P_{ts}(y = 0) = 0.70$ and $P_{ts}(y = 1) = 0.30$. This example is illustrated in the figure below

Training



Testing



[6] *Moreno-Torres et al., Pattern Recognition, 2011.*

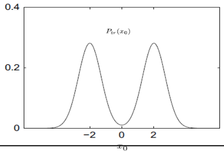
Prior probability shift: Example

Example [6]: Y → X problem with one covariate x_0 and a target y that may take the class value $y=0$ and $y=1$. In training data, $P_{tr}(y=0) = P_{tr}(y=1) = 0.5$ and $P_{tr}(x_0 \mid y)$ is defined as

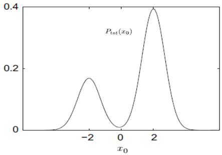
$$x_0 = \begin{cases} \mathcal{N}(2, 0.5), & \text{when } y = 1 \\ \mathcal{N}(-2, 05), & \text{otherwise} \end{cases} \tag{1}$$

Now consider that in the test data $P_{ts}(x_0 \mid y=0)$ and $P_{ts}(x_0 \mid y=1)$ remains unchanged, but the class prior probabilities vary, taking the values $P_{ts}(y=0) = 0.70$ and $P_{ts}(y=1) = 0.30$. This example is illustrated in the figure below

Training



Testing



[6] Moreno-Torres et al., Pattern Recognition, 2011.

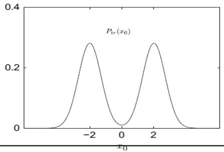
Prior probability shift: Example

Example [6]: $Y \rightarrow X$ problem with one covariate x_0 and a target y that may take the class value $y=0$ and $y=1$. In training data, $P_{tr}(y=0) = P_{tr}(y=1) = 0.5$ and $P_{tr}(x_0 \mid y)$ is defined as

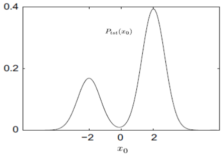
$$x_0 = \begin{cases} \mathcal{N}(2, 0.5), & \text{when } y = 1 \\ \mathcal{N}(-2, 05), & \text{otherwise} \end{cases} \tag{1}$$

Now consider that in the test data $P_{ts}(x_0 \mid y=0)$ and $P_{ts}(x_0 \mid y=1)$ remains unchanged, but the class prior probabilities vary, taking the values $P_{ts}(y=0) = 0.70$ and $P_{ts}(y=1) = 0.30$. This example is illustrated in the figure below

Training



Testing



[6] Moreno-Torres et al., Pattern Recognition, 2011.

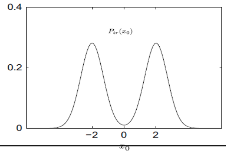
Prior probability shift: Example

Example [6]: $Y \rightarrow X$ problem with one covariate x_0 and a target y that may take the class value $y{=}0$ and $y{=}1$. In training data, $P_{tr}(y=0) = P_{tr}(y=1) = 0.5$ and $P_{tr}(x_0 \mid y)$ is defined as

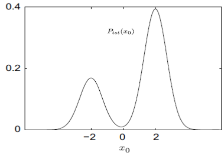
$$x_0 = \begin{cases} \mathcal{N}(2, 0.5), & \text{when } y = 1 \\ \mathcal{N}(-2, 05), & \text{otherwise} \end{cases} \tag{1}$$

Now consider that in the test data $P_{ts}(x_0 \mid y = 0)$ and $P_{ts}(x_0 \mid y = 1)$ remains unchanged, but the class prior probabilities vary, taking the values $P_{ts}(y = 0) = 0.70$ and $P_{ts}(y = 1) = 0.30$. This example is illustrated in the figure below

Training



Testing



[6] *Moreno-Torres et al., Pattern Recognition, 2011.*

Concept Shift

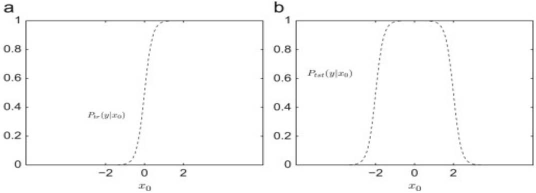
- Concept shift is defined as:
  - X $\rightarrow$ Y problems

$$P_{tr}(Y \mid X) \neq P_{ts}(Y \mid X) \quad \text{and} \quad P_{tr}(X) = P_{ts}(X)$$

&

  - Y $\rightarrow$ X problems

$$P_{tr}(X \mid Y) \neq P_{ts}(X \mid Y) \quad \text{and} \quad P_{tr}(Y) = P_{ts}(Y)$$



Concept Shift

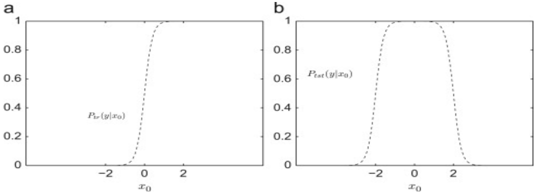
- Concept shift is defined as:
  - X → Y problems

$$P_{tr}(Y \mid X) \neq P_{ts}(Y \mid X) \quad \text{and} \quad P_{tr}(X) = P_{ts}(X)$$

&

  - Y → X problems

$$P_{tr}(X \mid Y) \neq P_{ts}(X \mid Y) \quad \text{and} \quad P_{tr}(Y) = P_{ts}(Y)$$



Concept Shift

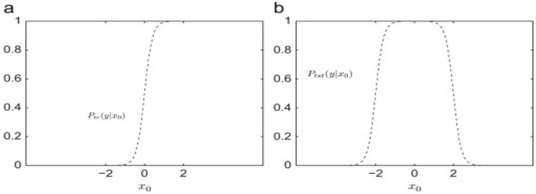
- Concept shift is defined as:
  - X → Y problems

$$P_{tr}(Y \mid X) \neq P_{ts}(Y \mid X) \quad \text{and} \quad P_{tr}(X) = P_{ts}(X)$$

&

  - Y → X problems

$$P_{tr}(X \mid Y) \neq P_{ts}(X \mid Y) \quad \text{and} \quad P_{tr}(Y) = P_{ts}(Y)$$



Outline

Shifts in Data

- Dataset Shift
- Types of Dataset Shift
- Causes of Dataset Shift

Learning in Dataset Shift: **Domain Adaptation**

Causes of Dataset Shift

1. The main two causes of dataset are Sample **Selection Bias** and **Non-stationary environments**.

Causes of Dataset Shift

1. The main two causes of dataset are Sample **Selection Bias** and **Non-stationary environments**.

Causes of Dataset Shift. . .

1. **Sample selection bias**: the discrepancy in distribution is due to the fact that the training examples have been obtained through a biased method, and thus do not represent reliably the operating environment where the classifier is to be deployed (In ML terms, would constitute the test set).
2. **Non-stationary environments**: It appears when the training environment is different from the test one, whether it is due to a temporal or a spatial change.

Causes of Dataset Shift...

1. **Sample selection bias**: the discrepancy in distribution is due to the fact that the training examples have been obtained through a biased method, and thus do not represent reliably the operating environment where the classifier is to be deployed (In ML terms, would constitute the test set).
2. **Non-stationary environments**: It appears when the training environment is different from the test one, whether it is due to a temporal or a spatial change.

Causes of Dataset Shift. . .

1. **Sample selection bias**: the discrepancy in distribution is due to the fact that the training examples have been obtained through a biased method, and thus do not represent reliably the operating environment where the classifier is to be deployed (In ML terms, would constitute the test set).
2. **Non-stationary environments**: It appears when the training environment is different from the test one, whether it is due to a temporal or a spatial change.

Sample selection bias

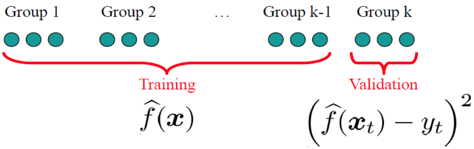
1. The term Sample selection bias refers to a systematic flaw in the process of data collection or labeling which causes training examples to be selected non-uniformly from the population to be modeled.
2. The term has been used as a synonym of covariate shift (which is not correct), but also on its own as a related problem to Dataset shift

Sample selection bias

1. The term Sample selection bias refers to a systematic flaw in the process of data collection or labeling which causes training examples to be selected non-uniformly from the population to be modeled.
2. The term has been used as a synonym of covariate shift (which is not correct), but also on its own as a related problem to Dataset shift

Correcting: Dataset shift generated by Sample Selection Bias

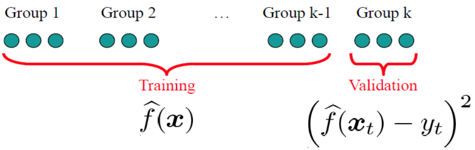
1. Divide the training samples into k groups.
2. Train a learning machine with $k-1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Correcting: Dataset shift generated by Sample Selection Bias

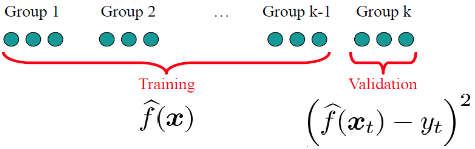
1. Divide the training samples into k groups.
2. Train a learning machine with $k-1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Correcting: Dataset shift generated by Sample Selection Bias

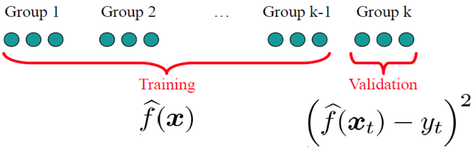
1. Divide the training samples into k groups.
2. Train a learning machine with $k-1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Correcting: Dataset shift generated by Sample Selection Bias

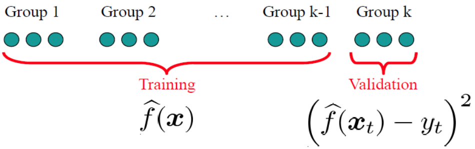
1. Divide the training samples into k groups.
2. Train a learning machine with $k-1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Correcting: Dataset shift generated by Sample Selection Bias

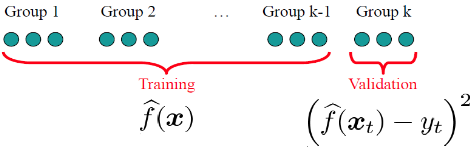
1. Divide the training samples into k groups.
2. Train a learning machine with $k-1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Correcting: Dataset shift generated by Sample Selection Bias

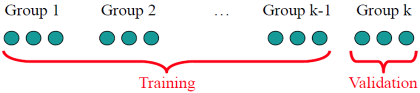
1. Divide the training samples into k groups.
2. Train a learning machine with $k-1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Correcting: Dataset shift generated by Sample Selection Bias

1. Divide the training samples into k groups.
2. Train a learning machine with $k - 1$ groups.
3. Validate the trained machine using the rest.
4. Repeat this for all the combination and output the mean validation error.



$$\widehat{f}(\boldsymbol{x}) \qquad \left(\widehat{f}(\boldsymbol{x}_t) - y_t\right)^2$$

5. This method is cross-validation (CV) and is almost unbiased without covariate shift.
6. But, CV is heavily biased under covariate shift.

Non-stationary environments

1. In real-world applications, it is often the case that the data is not (time- or space-) stationary
2. One of the most relevant non-stationary scenarios involves adversarial classification problems, such as spam filtering, fraud detection, and network intrusion detection.
3. This type of problem is receiving an increasing amount of attention in the machine learning field.

Non-stationary environments

1. In real-world applications, it is often the case that the data is not (time- or space-) stationary
2. One of the most relevant non-stationary scenarios involves adversarial classification problems, such as spam filtering, fraud detection, and network intrusion detection.
3. This type of problem is receiving an increasing amount of attention in the machine learning field.

Non-stationary environments

1. In real-world applications, it is often the case that the data is not (time- or space-) stationary
2. One of the most relevant non-stationary scenarios involves adversarial classification problems, such as spam filtering, fraud detection, and network intrusion detection.
3. This type of problem is receiving an increasing amount of attention in the machine learning field.

Non-stationary environments

1. In real-world applications, it is often the case that the data is not (time- or space-) stationary
2. One of the most relevant non-stationary scenarios involves adversarial classification problems, such as spam filtering, fraud detection, and network intrusion detection.
3. This type of problem is receiving an increasing amount of attention in the machine learning field.

Learning in Dataset Shift: **Domain Adaptation**

Learning in Non-stationary environments

Ditzler, G et al., (2015). Learning in Nonstationary Environments : A Survey. *IEEE Computational Intelligence Magazine*, 10(4), 12–25.

Learning in Non-stationary environments: Domain Adaptation

Domain Adaptation scenario arises when we aim at learning from a source data distribution (Training distribution) a well performing model on a different (but related) target data distribution (Testing distribution).

1. **Passive Approach**: continuously update the model over time (without requiring an explicit detection of the change)
2. **Active Approach**: rely on an explicit detection of the change in the data distribution to activate an adaptation mechanism

Learning in Non-stationary environments: Domain Adaptation

Domain Adaptation scenario arises when we aim at learning from a source data distribution (Training distribution) a well performing model on a different (but related) target data distribution (Testing distribution).

1. **Passive Approach**: continuously update the model over time (without requiring an explicit detection of the change)
2. **Active Approach**: rely on an explicit detection of the change in the data distribution to activate an adaptation mechanism

Passive Approaches

1. Does not uses a shift detection method to detect changes.
2. Perform a continuous adaptation of the model parameters every time new data arrive.
3. Advantage: Maintain an up-to-date model at all times.
4. Advantage: Avoiding the potential pitfall associated with the active approaches, that is, failing to detect a change or falsely detecting a non-existent change (false alarm).
5. Disadvantage: Update every time a new data arrives. Not suitable for real-time systems.

Passive Approaches

1. Does not uses a shift detection method to detect changes.
2. Perform a continuous adaptation of the model parameters every time new data arrive.
3. Advantage: Maintain an up-to-date model at all times.
4. Advantage: Avoiding the potential pitfall associated with the active approaches, that is, failing to detect a change or falsely detecting a non-existent change (false alarm).
5. Disadvantage: Update every time a new data arrives. Not suitable for real-time systems.

Passive Approaches

1. Does not uses a shift detection method to detect changes.
2. Perform a continuous adaptation of the model parameters every time new data arrive.
3. Advantage: Maintain an up-to-date model at all times.
4. Advantage: Avoiding the potential pitfall associated with the active approaches, that is, failing to detect a change or falsely detecting a non-existent change (false alarm).
5. Disadvantage: Update every time a new data arrives. Not suitable for real-time systems.

Passive Approaches

1. Does not uses a shift detection method to detect changes.
2. Perform a continuous adaptation of the model parameters every time new data arrive.
3. Advantage: Maintain an up-to-date model at all times.
4. Advantage: Avoiding the potential pitfall associated with the active approaches, that is, failing to detect a change or falsely detecting a non-existent change (false alarm).
5. Disadvantage: Update every time a new data arrives. Not suitable for real-time systems.

Passive Approaches

1. Does not uses a shift detection method to detect changes.
2. Perform a continuous adaptation of the model parameters every time new data arrive.
3. Advantage: Maintain an up-to-date model at all times.
4. Advantage: Avoiding the potential pitfall associated with the active approaches, that is, failing to detect a change or falsely detecting a non-existent change (false alarm).
5. Disadvantage: Update every time a new data arrives. Not suitable for real-time systems.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

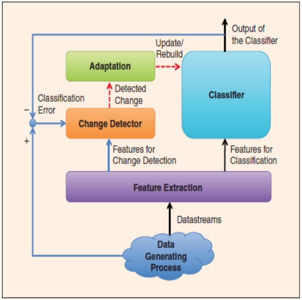
Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Passive Approaches Methods

1. **Single Classifier**:
   - Provide lower computational cost.
   - Decision trees are the most common classifier for data stream mining.
   - Very Fast Decision Tree (VFDT) and Online Information Network (ONI) are very popular approaches based on sliding window method.
   - Recently, Extreme Learning Machine (ELM) based on neural networks gaining popularity for learning non-stationary data.
2. **Ensemble Classifier**:
   - More accurate than single classifier due to reduction in the variance of the error.
   - Flexible to incorporate new data, simply by adding new members to ensemble.
   - Provide mechanism to forget irrelevant knowledge, simply by removing old classifiers.

Active Approaches Methods



It is based on change detection mechanism that triggers, whenever advisable, an adaptation mechanism aiming at reacting to the detected change by updating or building new classifier.

Active Approaches Methods

1. **Change/Shift Detection**:
   - Hypothesis Test, Change-point methods, Sequential hypothesis test, and Change-detection test.
   - Popular methods: EWMA, CUSUM, JIT, ICI, DDM and many more.
2. **Adaptation**:
   - Supervised adaptation, unsupervised adaptation, semi-supervised adaptation, and transduction.
   - Popular methods: Learn^{++}.NSE, COMPOSE, JIT adpative classifier, MOA, and many more.

Active Approaches Methods

1. **Change/Shift Detection**:
   - Hypothesis Test, Change-point methods, Sequential hypothesis test, and Change-detection test.
   - Popular methods: EWMA, CUSUM, JIT, ICI, DDM and many more.
2. **Adaptation**:
   - Supervised adaptation, unsupervised adaptation, semi-supervised adaptation, and transduction.
   - Popular methods: Learn^{++}.NSE, COMPOSE, JIT adpative classifier, MOA, and many more.

Active Approaches Methods

1. **Change/Shift Detection**:
   - Hypothesis Test, Change-point methods, Sequential hypothesis test, and Change-detection test.
   - Popular methods: EWMA, CUSUM, JIT, ICI, DDM and many more.
2. **Adaptation**:
   - Supervised adaptation, unsupervised adaptation, semi-supervised adaptation, and transduction.
   - Popular methods: Learn^{++}.NSE, COMPOSE, JIT adpative classifier, MOA, and many more.

Active Approaches Methods

1. **Change/Shift Detection**:
   - Hypothesis Test, Change-point methods, Sequential hypothesis test, and Change-detection test.
   - Popular methods: EWMA, CUSUM, JIT, ICI, DDM and many more.
2. **Adaptation**:
   - Supervised adaptation, unsupervised adaptation, semi-supervised adaptation, and transduction.
   - Popular methods: Learn^{++}.NSE, COMPOSE, JIT adpative classifier, MOA, and many more.

Active Approaches Methods

1. **Change/Shift Detection**:
   - Hypothesis Test, Change-point methods, Sequential hypothesis test, and Change-detection test.
   - Popular methods: EWMA, CUSUM, JIT, ICI, DDM and many more.
2. **Adaptation**:
   - Supervised adaptation, unsupervised adaptation, semi-supervised adaptation, and transduction.
   - Popular methods: Learn^{++}.NSE, COMPOSE, JIT adpative classifier, MOA, and many more.

Domain Adaptation

- Learning from a source (training) data distribution a well performing model on a different (but related) target (testing) data distribution.
- Example, one of the tasks of the **common spam filtering problem** consists in adapting a model from one user (the source distribution) to a new one who receives significantly different emails (the target distribution).
- Note that, when more than one source distribution is available the problem is referred to as **multi-source domain adaptation.**
- Iterative Domain Adaptation Algorithm
  1. a model h is learned from the labeled examples;
  2. h automatically labels some target examples;
  3. a new model is learned from the new labeled examples.

Domain Adaptation

- Learning from a source (training) data distribution a well performing model on a different (but related) target (testing) data distribution.
- Example, one of the tasks of the **common spam filtering problem** consists in adapting a model from one user (the source distribution) to a new one who receives significantly different emails (the target distribution).
- Note that, when more than one source distribution is available the problem is referred to as **multi-source domain adaptation.**
- Iterative Domain Adaptation Algorithm
  1. a model h is learned from the labeled examples;
  2. h automatically labels some target examples;
  3. a new model is learned from the new labeled examples.

Domain Adaptation

- Learning from a source (training) data distribution a well performing model on a different (but related) target (testing) data distribution.
- Example, one of the tasks of the **common spam filtering problem** consists in adapting a model from one user (the source distribution) to a new one who receives significantly different emails (the target distribution).
- Note that, when more than one source distribution is available the problem is referred to as **multi-source domain adaptation.**
- Iterative Domain Adaptation Algorithm
    1. a model h is learned from the labeled examples;
    2. h automatically labels some target examples;
    3. a new model is learned from the new labeled examples.

Domain Adaptation

- Learning from a source (training) data distribution a well performing model on a different (but related) target (testing) data distribution.
- Example, one of the tasks of the **common spam filtering problem** consists in adapting a model from one user (the source distribution) to a new one who receives significantly different emails (the target distribution).
- Note that, when more than one source distribution is available the problem is referred to as **multi-source domain adaptation.**
- Iterative Domain Adaptation Algorithm
  1. a model h is learned from the labeled examples;
  2. h automatically labels some target examples;
  3. a new model is learned from the new labeled examples.

Domain Adaptation

- Learning from a source (training) data distribution a well performing model on a different (but related) target (testing) data distribution.
- Example, one of the tasks of the **common spam filtering problem** consists in adapting a model from one user (the source distribution) to a new one who receives significantly different emails (the target distribution).
- Note that, when more than one source distribution is available the problem is referred to as **multi-source domain adaptation.**
- Iterative Domain Adaptation Algorithm
  1. a model h is learned from the labeled examples;
  2. h automatically labels some target examples;
  3. a new model is learned from the new labeled examples.

Domain Adaptation

- Learning from a source (training) data distribution a well performing model on a different (but related) target (testing) data distribution.
- Example, one of the tasks of the **common spam filtering problem** consists in adapting a model from one user (the source distribution) to a new one who receives significantly different emails (the target distribution).
- Note that, when more than one source distribution is available the problem is referred to as **multi-source domain adaptation.**
- Iterative Domain Adaptation Algorithm
  1. a model h is learned from the labeled examples;
  2. h automatically labels some target examples;
  3. a new model is learned from the new labeled examples.

Summary

- Why is it difficult to learn from Data.
- Dataset shift and types of dataset shift.
- Causes of dataset shift.
- How to handle sample selection bias.
- Approaches to non-stationary learning (i.e. Passive and Actives Approaches).

Summary

- Why is it difficult to learn from Data.
- Dataset shift and types of dataset shift.
- Causes of dataset shift.
- How to handle sample selection bias.
- Approaches to non-stationary learning (i.e. Passive and Actives Approaches).

Summary

- Why is it difficult to learn from Data.
- Dataset shift and types of dataset shift.
- Causes of dataset shift.
- How to handle sample selection bias.
- Approaches to non-stationary learning (i.e. Passive and Actives Approaches).

Summary

- Why is it difficult to learn from Data.
- Dataset shift and types of dataset shift.
- Causes of dataset shift.
- How to handle sample selection bias.
- Approaches to non-stationary learning (i.e. Passive and Actives Approaches).

Summary

- Why is it difficult to learn from Data.
- Dataset shift and types of dataset shift.
- Causes of dataset shift.
- How to handle sample selection bias.
- Approaches to non-stationary learning (i.e. Passive and Actives Approaches).

Contact me:

h.raza@essex.ac.uk
sagihaider@gmail.com

Webpage:

https://sites.google.com/site/whoishraza/home
https://www.essex.ac.uk/people/razah72409

Follow me:

: @sagihaider

Linkedin : sagihaider

THANK YOU!